How to / Trackmania / Trackmaking / Blockmix / Hex-Edit

Author : NL_prof , TStarGermany (Translations: Natzor _ DexM)

PAGE 1 2
NEXT PAGE

Introduction

Qu'entend-on par "Blockmixing" ou "Hex editing" ? C'est simplement l'art de fusionner deux ou plusieurs parties d'un circuit ne pouvant normalement pas l'être, ou l'art de placer des blocs là où il est théoriquement impossible de le faire.

Le résultat ? Des circuits hors du commun et des effets visuels exceptionnels ouvrant la porte à des possibilités que n'offre pas l'éditeur interne du jeu.

Les techniques de base du "Blockmixing" sont faciles à maîtriser, mais les techniques plus avancées requièrent un certain entraînement et beaucoup de patience. Bonne chance à tous dans l'apprentissage de ce nouvel art :-)

Table of contents

Partie 1 : Structure d'un fichier de circuit (GBX)
-- Décimal & Hexadécimal
-- Placement et stockage des blocs
Partie 2 : Blockmixing manuel
Partie 3 : Blockmixing moderne
-- Placement de blocs dans l'éditeur
-- Avec ChallengeEdit
-- Le processus de guérison
-- Finition
Partie 4 : Conseils et astuces
Partie 5 : Fonctions avancées de ChallengeEdit
Partie 6 : Structures et fonctions spéciales

Partie 1 : Structure d'un fichier de circuit (GBX)

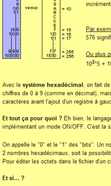
Les circuits de Trackmania se présentent sous la forme de fichiers ".GBX", format propre à Nadeo. Un fichier .GBX contient de nombreuses données différentes, écrites en "langage machine", dont la plupart sont cryptées et/ou compressées. L'outil "Hex-Editor" permet de lire et/ou modifier ces fichiers.

Le fichier .GBX d'un circuit contient essentiellement les informations suivantes:
* les informations générales du circuit (telles que son nom, le nom de l'auteur, le temps de validation, le mot de passe, etc.)
* l'emplacement des différents blocs composant votre circuit
* le contenu du Mediatracker
* les chemins d'accès dont dépend le circuit (emplacements/URLs des signaux, "mods", musiques, etc.)
* une miniature du circuit au format JPG
* et quelques autres infos...

Ces différentes parties sont surlignées en couleur dans l'image ci-contre...

Comme vous pouvez le constater, chaque information s'affiche en code hexadécimal sur la gauche...

Décimal & Hexadécimal



Il est vrai qu'on n'y pense pas souvent, mais notre façon de compter est relativement complexe !

Dans notre **système décimal** "normal", nous comptons avec les chiffres de 0 à 9... Si nous voulons continuer à compter, nous ajoutons un "registre" à gauche, nous incrémentons sa valeur à "1" et poursuivons le comptage à droite...

Par exemple :
576 signifie en fait: 5(76 (100*5 + 10*7 + 1*6)

Ou plus précisément :
10^2*5 + 10^1*7 + 10^0*6

Avec le **système hexadécimal**, on fait de même, à l'exception du fait qu'on ne se contente plus d'utiliser seulement les chiffres de 0 à 9 (comme en décimal), mais également les lettres de A à F... Ce qui implique un comptage de 16 caractères avant l'ajout d'un registre à gauche et le poursuite du comptage...

Et tout ça pour quoi ? Eh bien, le langage machine se base sur deux valeurs, "0" et "1", deux signaux électriques implémentant un mode ON/OFF. C'est la seule forme d'information que les ordinateurs comprennent.

On appelle le "0" et le "1" des "bits". Un nombre hexadécimal représente 2^4 bits. Un "octet" est représenté par 2^8 bits = 2 nombres hexadécimaux, soit la possibilité de prendre 256 valeurs différentes.
Pour éditer les octets dans le fichier d'un circuit, nous devons pouvoir lire & écrire ces octets.

Et si... ?

Et si la valeur qu'on désire donner dépasse 256 ? Alors il vous faut un octet additionnel qu'on rattache au premier (d'où les valeurs multi-octets)... Mais pas de panique, TM n'utilise pas les valeurs multi-octets :-)

Quelle valeur hexadécimale représente quel caractère "réel" ?
Les octets (nombres hexadécimaux) à gauche sont principalement représentés par des caractères "ASCII" à droite. Les quelques blancs sont dus au fait que certains octets ne peuvent être représentés par des caractères ASCII.

Cette introduction n'est pas très exhaustive , mais elle a avant tout pour but de vous aider à mieux cerner ce que les valeurs hexadécimales de l'outil "Hex Editor" représentent. Il ne faut pas oublier non plus que les fichiers .GBX servent également pour les replays et les modèles de voiture 3D... mais nous n'aborderons ici que les fichiers représentant des circuits, puisque c'est le "Blockmixing" qui nous intéresse.

Placement et stockage des blocs

Dans le cas de l'environnement Stadium, vous avez 32*32*32 cellules où placer vos blocs. Chaque bloc que vous placez possède un nom et des coordonnées (longueur, hauteur et profondeur, prenant des valeurs de 0 à 31).

Dans le cas de la hauteur (Y dans l'image), l'herbe se trouve au niveau "0". Tout bloc placé directement sur l'herbe est donc au niveau "1".

En plus du nom du bloc et de sa **position XYZ**, sa rotation et ses variations sont stockées dans le fichier du circuit (les **variations** entrent en jeu quand un même bloc prend différentes formes, suivant s'il est placé directement sur l'herbe, connecté ou non avec d'autres blocs).

Partie 2 : Blockmixing manuel

Il n'y a pas si longtemps, le "Blockmixing" devait se faire avec l'outil "Hex Editor". Si cette partie ne vous intéresse pas, vous pouvez directement sauter à la Partie 3.

A des fins d'optimisation (taille des fichiers), les fichiers .GBX de Nadeo compressent toutes les informations concernant l'emplacement des différents blocs du circuit. Ceci rend virtuellement impossible la recherche d'une information pour un bloc en particulier que vous auriez souhaité modifier.

Plaçons un checkpoint (sur l'herbe) aux coordonnées **X:17/ Y:1/ Z:17**, juste à droite du turbo avec lequel vous souhaitez fusionner le checkpoint.

Il s'agit donc de déplacer le checkpoint de **X:17 en X:16**. Tout le reste demeure inchangé.

Un logiciel, nommé "**ReCompressor**", nous permet de décompresser l'information portée par un fichier de circuit, de sauvegarder le fichier sous cette forme et de **lister l'ensemble des blocs, ainsi que leurs adresses hexadécimales** (où les trouver).

Vous pouvez maintenant ouvrir votre circuit décompressé avec l'outil "Hex Editor".

1) Commencez par aller à **l'adresse hexadécimale 00001760** (que vous connaissez grâce à la liste du "ReCompressor"), symbolisée par un carré rouge.
2) A droite, vous pouvez voir où le "**StadiumGrassCheckPoint**" (en jaune) commence.

3) Après la description du bloc se trouvent **plusieurs octets**, portant l'information des **coordonnées**, **de la rotation** et **des variations** du bloc. L'image vous aide à comprendre ce à quoi chaque octet correspond.

4) Nous voulons déplacer le checkpoint de **X:17 en X:16**. Il faut donc modifier l'octet correspondant (cercle bleu foncé) de la **valeur hexadécimale "11" à "10"** (représentant les valeurs "réelles" (décimales) "17" et "16" de nos coordonnées). On sauvegarde alors le fichier et on retourne dans TM.

En revenant le circuit dans TM, on s'aperçoit que le checkpoint s'est déplacé d'une cellule, pile sur le turbo :-)

On peut maintenant poursuivre la construction du circuit.

Heureusement, TM fera à nouveau appel à sa méthode de compression de données aussitôt après la première sauvegarde du circuit dans l'éditeur.

Voilà en quelques mots comment fonctionne le "Blockmixing" manuel. Avant l'existence de l'outil "ReCompressor", la tâche était d'autant plus ardue.

Néanmoins, à l'heure actuelle, nous vous déconseillons d'utiliser l'outil "ReCompressor" si vous faites un "Blockmixing" manuel, l'algorithme de compression des fichiers .GBX laissant çà et là des erreurs (groupements, masquages ou omissions d'octets) dans certains blocs.

Nous vous recommandons plutôt l'outil "ChallengeEdit" pour la décompression de vos fichiers, comme il est expliqué ci-dessous.

Partie 3 : Blockmixing moderne

Heureusement pour nous, le "Blockmixing" s'est vite répandu, si bien qu'il existe aujourd'hui des méthodes nettement plus simples pour fusionner deux blocs.

Des membres du forum francophone TM-Hexa ont mis à notre disposition d'excellents programmes (tels que "**ChallengeEdit**") permettant de décompresser un fichier de circuit et de lister ses composants. L'édition hexadécimale se fait d'elle-même. Au final, un superbe outil multifonctions pour le "Blockmixing".

ChallengeEdit ne peut éditer que des circuits dont l'auteur est l'un des profils en ligne sauvegardés sur votre ordinateur. Donc, avant de pouvoir éditer nos fichiers d'exemple, il vous faut les ouvrir dans l'éditeur et les réenregistrer, de manière à ce que "vous" soyez l'auteur du fichier.

Placement de blocs dans l'éditeur

Cliquez sur l'icône pour télécharger le fichier "**TMC-Blockmix-Example1**" et mettez-le dans votre dossier "Tracks/Challenges". Ouvrez-le et sauvez-le dans l'éditeur, afin de pouvoir l'utiliser avec ChallengeEdit.

Ouvrez le fichier dans l'éditeur de TM. Nous souhaitons construire une route avec un virage en épingle à cheveux.

Placez la ligne de départ, une petite section de route sinueuse et la ligne d'arrivée.

A présent, sélectionnez le premier bloc "virage" disponible (2/1/2). Afin de ne pas interférer avec les autres blocs, il nous faut le placer 2 cellules au-dessus du sol (en Y=3). Sauvez le circuit.

Avec ChallengeEdit

Lancez ChallengeEdit et chargez votre circuit.

Etape1) "StadiumRoadMainGTCurve2" est le bloc que l'on désire abaisser au niveau du sol. Sélectionnez-le avec la souris.

Etape 2) En bas, vous pouvez maintenant voir toutes les valeurs importantes pour ce bloc. Changez la valeur de "Z" de "3" à "1" (vous l'abaissez de 2 cellules) à l'aide des flèches sur la droite (ou directement en tapant "1" sur le clavier).

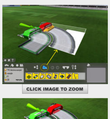
Remarque : ChallengeEdit utilise les coordonnées XZY (au lieu de XYZ), mais c'est sans importance...
Rendez-vous LÀ pour une explication détaillée de la raison pour laquelle les variables sont inversées.

Etape 3) Cliquez maintenant sur le bouton "Save block modification...".

Etape 4) Cliquez sur le bouton "Save track".

L'édition hexadécimale s'est faite d'elle-même. Retournez dans TM et réouvrez le circuit dans l'éditeur.

Le processus de "guérison"



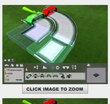
Une fois retournés dans l'éditeur TM, nous voilà confrontés à un problème : nous avons déplacé un "virage en l'air" au niveau du sol, ce qui nous donne un joli trou d'air dans le sol :-)

C'est là qu'intervient le "**processus de guérison**". Nous utilisons l'éditeur de TM pour réparer l'erreur...
Cliquez sur le bouton "Supprimer" et cliquez sur le virage.

Une fois le bloc "virage" supprimé, le sol réapparaît, mais tout a l'air un peu embrouillé...

Cliquez sur le bouton "**Annulation**" et TM va tenter de replacer le bloc, mais cette fois en faisant en sorte que le bloc soit correctement connecté à ses voisins, ainsi qu'au sol.

Finition



Il ne nous reste plus qu'à relier notre virage à la partie gauche du circuit avec une nouvelle section de route (2/1/1).

Voilà notre virage en épingle à cheveux terminé !

Ceci n'est qu'un exemple très simple du "Blockmixing/Hex editing" moderne. Mais la plupart des mixages fonctionnent comme cela.

Cliquez sur l'icône pour télécharger le fichier "**TMC-Blockmix-Example1-Mixed**". C'est la version mixée de l'exemple , il ne vous reste plus qu'à le "guérir" comme expliqué ci-dessus.

PAGE 1 2
NEXT PAGE

This site is friendly to stationary (PC/Mac) and mobile (Android/iOS) browsers
Trackmania, Shootmania, Quantmania games & media © Nadeo & Ubisoft used under fair use policy
8,211,239 page views since January 2006. All tutorials, services, coding © Irene Creative 2006-2013